Group-theoretic Algorithms for Matrix Multiplication

Christopher Umans
Computer Science Department
California Institute of Technology
1200 E. California Blvd.
Pasadena, CA 91125
umans@cs.caltech.edu

INVITED TALK ABSTRACT

The *exponent of matrix multiplication* is the smallest real number ω such that for all $\epsilon > 0$, $O(n^{\omega+\epsilon})$ arithmetic operations suffice to multiply two $n \times n$ matrices. The standard algorithm for matrix multiplication shows that $\omega \leq 3$. Strassen's remarkable result [5] shows that $\omega \leq 2.81$, and a sequence of further works culminating in the work of Coppersmith and Winograd [4] have improved this upper bound to $\omega \leq 2.376$ (see [1] for a full history). Most researchers believe that in fact $\omega = 2$, but there have been no further improvements in the known upper bounds for the past fifteen years.

It is known that several central linear algebra problems (for example, computing determinants, solving systems of equations, inverting matrices, computing LUP decompositions) have the same exponent as matrix multiplication, which makes ω a fundamental number for understanding algorithmic linear algebra. In addition, there are non-algebraic algorithms whose complexity is expressed in terms of ω.

In this talk I will describe a new "group-theoretic" approach, proposed in [3], to devising algorithms for fast matrix multiplication. The basic idea is to reduce matrix multiplication to group algebra multiplication with respect to a suitable non-abelian group. The group algebra multiplication is performed in the Fourier domain, and then using this scheme recursively yields upper bounds on ω.

This general framework produces nontrivial matrix multiplication algorithms if one can construct finite groups with certain properties. In particular, a very natural embedding of matrix multiplication into $\mathbb{C}[G]$-multiplication is possible when group G has three subgroups H_1, H_2, H_3 that satisfy the *triple product property*. I'll define this property and describe a construction that satisfies the triple product property with parameters that are necessary (but not yet sufficient) to achieve $\omega = 2$.

In the next part of the talk I'll describe demands on the representation theory of the groups in order for the overall approach to yield non-trivial bounds on ω, namely, that the character degrees must be "small." Constructing families of groups together with subgroups satisfying the triple product property *and* for which the character degrees are sufficiently small has turned out to be quite challenging.

In [2], we succeed in constructing groups meeting both requirements, resulting in non-trivial algorithms for matrix multiplication in this framework. I'll outline the basic construction, together with more sophisticated variants that achieve the bounds $\omega < 2.48$ and $\omega < 2.41$.

In the final part of the talk I'll present two appealing conjectures, one combinatorial and the other algebraic. Either one would imply that the exponent of matrix multiplication is 2.

ISSAC'06, July 9–12, 2006, Genova, Italy.
ACM 1-59593-276-3/06/0007.

Categories and Subject Descriptors

I.1.2 [**Symbolic and Algebraic Manipulation**]: Algorithms

General Terms

Algorithms

Keywords

matrix multiplication, finite groups, representation theory

1. REFERENCES

[1] P. Bürgisser, M. Clausen, and M. A. Shokrollahi. *Algebraic Complexity Theory*. Springer-Verlag, 1997.

[2] H. Cohn, R. Kleinberg, B. Szegedy, and C. Umans. Group-theoretic algorithms for matrix multiplication. In *Proceedings of the 46th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 379–388. IEEE Computer Society, October 2005.

[3] H. Cohn and C. Umans. A group-theoretic approach to fast matrix multiplication. In *Proceedings of the 44th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 438–449. IEEE Computer Society, October 2003.

[4] D. Coppersmith and S. Winograd. Matrix multiplication via arithmetic progressions. *J. Symbolic Computation*, 9:251–280, 1990.

[5] V. Strassen. Gaussian elimination is not optimal. *Numerische Mathematik*, 13:354–356, 1969.